Cui_Jingyi_Assgn03

Jingyi Cui

February 13, 2020

Problem 1

(a)

```
dir <- "C:/Users/cuijy/Desktop"
IPO <- read.csv(file.path(dir, "IPOData_CFRM425.csv"))
fit01 = lm(rtn261 ~ DaysBetterThanSP + daysProfit+ volumeDay0 +
             volumeDay120 + rtn120 + MarketCap + CEOAge
           + NetIncome + employees, data = IPO)
summary(fit01)
#Coefficients:
#                   Estimate  t value Pr(>|t|)
#(Intercept)      -5.647e-01   -5.631 2.11e-08 ***
#DaysBetterThanSP  7.377e-04    1.980   0.0478 *
#daysProfit        1.939e-03   11.539  < 2e-16 ***
#volumeDay0        2.539e-11    0.498   0.6185
#volumeDay120      8.758e-10    0.151   0.8803
#rtn120            7.442e-01   22.584  < 2e-16 ***
#MarketCap         4.229e-13    0.904   0.3663
#CEOAge            2.487e-03    1.552   0.1207
#NetIncome         3.360e-13    0.669   0.5036
#employees         3.987e-07    0.927   0.3542
# Adjusted R-squared:  0.5169
AIC(fit01)
#> AIC(fit01)
#[1] 2269.222
```

(b)

```
# From the summary(fit01), we can observe that p-value for
# DaysBetterThanSP, dayProfit, rtn120 are small than 0.05,
# they are significant.
fit02 = lm(rtn261 ~ DaysBetterThanSP + daysProfit +
             rtn120 + rtn21 + rtn65, data = IPO)
summary(fit02)
#Coefficients:
#                   Estimate t value Pr(>|t|)
#(Intercept)      -0.4142491  -9.218   <2e-16 ***
#DaysBetterThanSP  0.0006101   1.673   0.0945 .
#daysProfit        0.0020113  11.983   <2e-16 ***
#rtn120            0.8468339  20.419   <2e-16 ***
#rtn21            -0.1386599  -2.451   0.0143 *
```

```
#rtn65            -0.1036841  -1.944   0.0521 .
AIC(fit02)
# Adjusted R-squared is 0.5219 and AIC value is 2251.484. The adjusted r squared increases,
# AIC decreses, implying added variables giving better predictive power.
```

(c)

```
# From summary(fit02), we can observe that p-value for
# daysProfit, rtn120, rtn21 are less than 0.05, they are significant.
fit03 = lm(rtn261 ~ daysProfit +
             rtn120 + rtn21 + as.factor(Sector)-1, data = IPO)
summary(fit03)
# Coefficients:
#                                      Estimate  t value Pr(>|t|)
#daysProfit                            0.001950   11.749  < 2e-16 ***
#rtn120                                0.806657   22.801  < 2e-16 ***
#rtn21                                -0.188740   -3.864 0.000116 ***
#as.factor(Sector)Basic Industries    -0.319824   -5.239 1.83e-07 ***
#as.factor(Sector)Capital Goods       -0.241371   -4.247 2.29e-05 ***
#as.factor(Sector)Consumer Durables   -0.349347   -4.037 5.68e-05 ***
#as.factor(Sector)Consumer Non-Durables -0.253522   -3.798 0.000152 ***
#as.factor(Sector)Consumer Services   -0.376032   -9.569  < 2e-16 ***
#as.factor(Sector)Energy              -0.229015   -3.939 8.53e-05 ***
#as.factor(Sector)Finance             -0.285078   -7.110 1.74e-12 ***
#as.factor(Sector)Health Care         -0.394615   -11.087 < 2e-16 ***
#as.factor(Sector)Miscellaneous       -0.366854   -5.480 4.92e-08 ***
#as.factor(Sector)Public Utilities    -0.357947   -5.061 4.65e-07 ***
#as.factor(Sector)Technology          -0.377268   -8.967  < 2e-16 ***
#as.factor(Sector)Transportation      -0.448938   -5.507 4.24e-08 ***
AIC(fit03)
# Adjusted R-squared increses to 0.524 and AIC value increases to 2252.507,
# leading worse predictive power, but better explanatory power to observed data.
```

Problem 2

(a)

```
rm(list = ls())
library(quantmod)
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo
```

```
## Loading required package: TTR
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

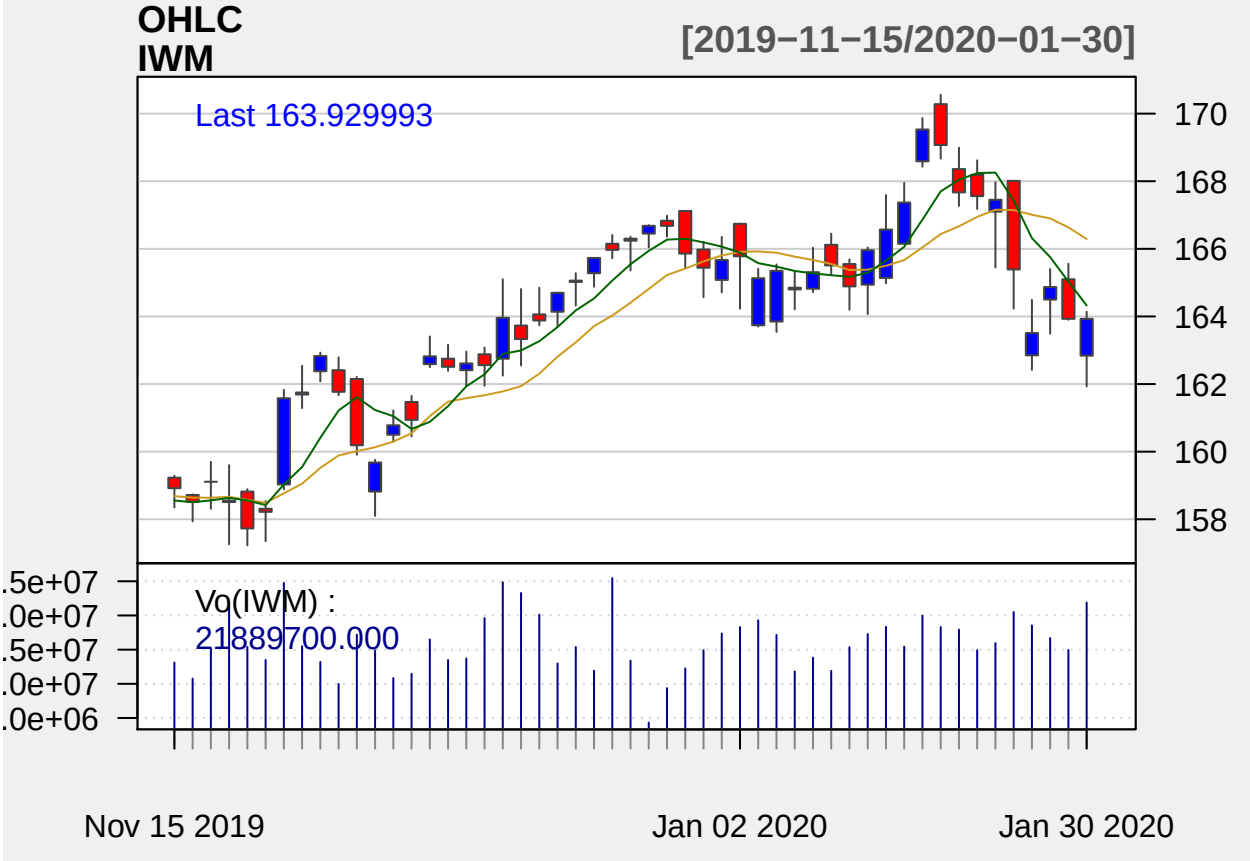
```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
getSymbols("IWM", from = "2001-01-01", to = "2020-01-31")
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

(b)

```
addTA(Vo(IWM), col = "darkblue", type = "h")
```




```
# 11/22, 12/6, 12/9, 1/2, 1/13, 1/24
```

(c)

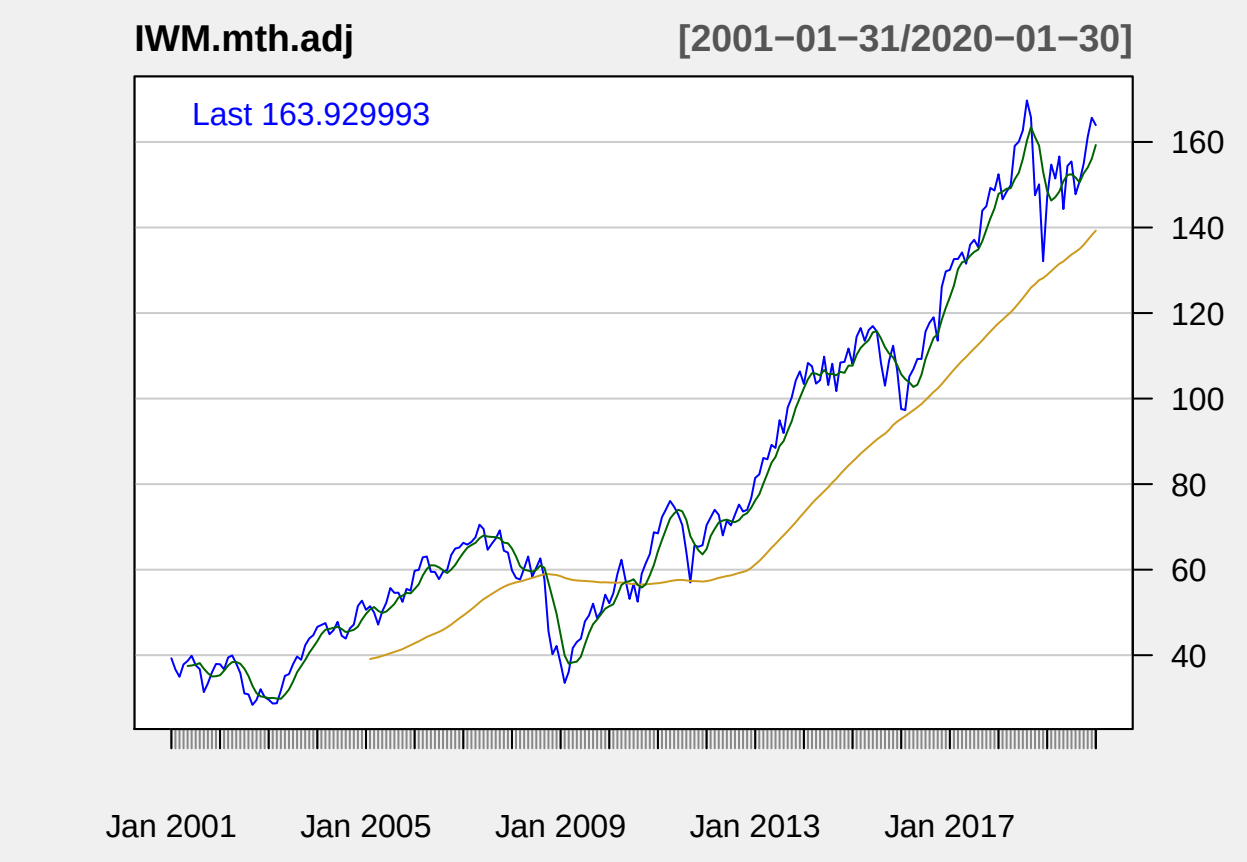
```r
IWM.mth <- to.period(IWM, "months")
IWM.mth.adj <- Ad(IWM.mth)
IWM.mth.adj[c(1:3),]
#> IWM.mth.adj[1,]
#            IWM.Adjusted
#2001-01-31     39.28318
#2001-02-28     36.65522
#2001-03-30     34.95457
last(IWM.mth.adj, 3)
#            IWM.Adjusted
#2019-11-29     161.1787
#2019-12-31     165.6700
#2020-01-30     163.9300
# Last trading day
```

For (ii) part, last trading is used.

```r
chartSeries(IWM.mth.adj, theme = chartTheme("white", up.col = "blue"), major.ticks = "year")
```

```r
addSMA(n = 50, col = "goldenrod3")
```

```r
addSMA(n = 5, col = "darkgreen")
```




```r
# The trend of the prices will change in a while; when the green line crossed
# the yellow line from above, implying decreasing price in the future.
```

(d)

```
IWM.Rtns <- diff(log(IWM.mth.adj), lag = 1)
IWM.Rtns <- IWM.Rtns[-1]
IWM.Rtns[c(1:3),]
#> IWM.Rtns[1,]
#           IWM.Adjusted
#2001-02-28  -0.01904269
#2001-03-30  -0.04750671
#2001-04-30   0.07935522
tail(IWM.Rtns, 3)
#           IWM.Adjusted
#2019-11-29  0.007871820
#2019-12-31  0.005393010
#2020-01-30 -0.002068358
```

(e)

```
IWMRtns <- IWM.Rtns["2008-01-01::2009-12-31"]

getSymbols("SPY", from = "2001-01-01", to = "2020-01-31")
SPY.mth <- to.period(SPY, "months")
head(SPY.mth, 3)
#           SPY.Open SPY.High  SPY.Low SPY.Close SPY.Volume SPY.Adjusted
#2001-01-31   132.00   138.70 127.5625    137.02  181296400     95.18305
#2001-02-28   137.10   137.99 121.8000    123.95  178607000     86.10375
#2001-03-30   124.05   127.75 108.0400    116.69  318187200     81.27876
SPY.mth.adj <- Ad(SPY.mth)
head(SPY.mth.adj, 3)
#           SPY.Adjusted
#2001-01-31     95.18305
#2001-02-28     86.10375
#2001-03-30     81.27876
SPY.Rtns <- diff(log(SPY.mth.adj), lag = 1)
SPY.Rtns <- SPY.Rtns[-1]
head(SPY.Rtns, 3)
#          SPY.Adjusted
#2001-02-28  -0.10024899
#2001-03-30  -0.05766814
#2001-04-30   0.08198544
SPYRtns <- SPY.Rtns["2008-01-01::2009-12-31"]
head(SPYRtns, 3)
#           SPY.Adjusted
#2008-01-31 -0.06236626
#2008-02-29 -0.02618207
#2008-03-31 -0.00898257

getSymbols("XLU", from = "2001-01-01", to = "2020-01-31")
XLU.mth <- to.period(XLU, "months")
head(XLU.mth, 3)
#           XLU.Open XLU.High XLU.Low XLU.Close XLU.Volume XLU.Adjusted
#2001-01-31  32.9375 33.92188 30.0625     31.32     973100     15.86927
#2001-02-28  31.4500 32.71000 31.2000     31.97     532900     16.19861
#2001-03-30  31.8000 32.06000 28.3000     31.29    1341800     15.96640
```

```
XLU.mth.adj <- Ad(XLU.mth)
head(XLU.mth.adj, 3)
#           XLU.Adjusted
#2001-01-31     15.86927
#2001-02-28     16.19861
#2001-03-30     15.96640
XLU.Rtns <- diff(log(XLU.mth.adj), lag = 1)
XLU.Rtns <- XLU.Rtns[-1]
head(XLU.Rtns, 3)
#           XLU.Adjusted
#2001-02-28   0.02054109
#2001-03-30  -0.01443917
#2001-04-30   0.04713020
XLURtns <- XLU.Rtns["2008-01-01::2009-12-31"]
head(XLURtns, 3)
#           XLU.Adjusted
#2008-01-31  -0.07630923
#2008-02-29  -0.04191682
#2008-03-31   0.01592532

getSymbols("XLF", from = "2001-01-01", to = "2020-01-31")
XLF.mth <- to.period(XLF, "months")
head(XLF.mth, 3)
#           XLF.Open XLF.High  XLF.Low XLF.Close XLF.Volume XLF.Adjusted
#2001-01-31 24.04041 24.90353 22.74573  23.90739   28694200     12.49948
#2001-02-28 24.00487 24.27295 21.26726  22.29894   13816600     11.65854
#2001-03-30 22.13648 23.10317 18.99269  21.55971   19338400     11.30934
XLF.mth.adj <- Ad(XLF.mth)
head(XLF.mth.adj, 3)
#           XLF.Adjusted
#2001-01-31     12.49948
#2001-02-28     11.65854
#2001-03-30     11.30934
XLF.Rtns <- diff(log(XLF.mth.adj), lag = 1)
XLF.Rtns <- XLF.Rtns[-1]
head(XLF.Rtns, 3)
#           XLF.Adjusted
#2001-02-28  -0.06964810
#2001-03-30  -0.03041003
#2001-04-30   0.03553269
XLFRtns <- XLF.Rtns["2008-01-01::2009-12-31"]
head(XLFRtns, 3)
#           XLF.Adjusted
#2008-01-31  0.007232978
#2008-02-29 -0.120575436
#2008-03-31 -0.029842055
mergeRtns <- merge(IWMRtns, SPYRtns, XLURtns, XLFRtns)
sapply(mergeRtns, FUN = sd)
#IWM.Adjusted SPY.Adjusted XLU.Adjusted XLF.Adjusted
#  0.08697567   0.06960655   0.05871521   0.13133530
```

(f)

```r
IWM.adj <- Ad(IWM.mth)["2008-01-01::2009-12-31"]
SPY.adj <- Ad(SPY.mth)["2008-01-01::2009-12-31"]
XLU.adj <- Ad(XLU.mth)["2008-01-01::2009-12-31"]
XLF.adj <- Ad(XLF.mth)["2008-01-01::2009-12-31"]
IWM.cum <- IWM.adj[,1]/as.numeric(IWM.adj[1,1])
SPY.cum <- SPY.adj[,1]/as.numeric(SPY.adj[1,1])
XLU.cum <- XLU.adj[,1]/as.numeric(XLU.adj[1,1])
XLF.cum <- XLF.adj[,1]/as.numeric(XLF.adj[1,1])
cumRtns <- merge(IWM.cum, SPY.cum, XLU.cum, XLF.cum)
zooRtns <- as.zoo(cumRtns)
plot(zooRtns, ylab = "Cumulative Return", xlab = "Time"
     , main = "ETF Cumulative Returns",
     col = c("red", "darkgreen", "goldenrod3", "darkblue"), screen = 1)
legend(x = "bottomleft", legend = c("IWM", "SPY", "XLU", "XLF")
       , lty = 1, col = c("red", "darkgreen", "goldenrod3", "darkblue"))
```